

Справочник по PHP и JavaScript

PHP	JavaScript
Переменные	
<pre>\$a = 'строка'; \$b = "строка"; \$c = 25; \$d = 1.25;</pre>	<pre>var a = 'строка'; var b = "строка"; var c = 25; var d = 1.25; let x = 100; // ES-2015</pre>
Константы	
<pre>define("name", "value"); const MIN_VALUE = 0.0;</pre>	<pre>const MIN_VALUE = 5; // ES-2015</pre>
Условия	
<pre>\$result = (value > 0) ? 'Yes' : 'No'; if(value > 10){ ... }elseif(value > 0) { ... }else { ... }</pre>	<pre>var result = value > 0 ? 'yes' : 'no'; if(value > 10){ ... }else if(value > 0) { ... }else { ... }</pre>
Циклы	
<pre>\$i = 0; while (\$i < 10){ \$i++; }</pre>	<pre>var i = 0; while (i < 10){ i++; }</pre>
<pre>do { \$i++; } while (\$i < 10);</pre>	<pre>do { i++; } while (i < 10);</pre>
<pre>for (\$i = 0; \$i < 10; \$i++){ print_r(\$i); }</pre>	<pre>for (var i = 0; i < 10; i++){ console.log(i); }</pre>
<pre>\$arr = array(1,2,3,4); foreach (\$arr as \$key => \$value){ echo \$key." : ".\$value; } foreach (\$arr as &\$value){ \$value = \$value * 2; }</pre>	<pre>for(var key in obj) console.log(obj[key]); var arr = [1,2,3,4,5]; arr.forEach(function(item, i, arr){ alert(i + " : " + item); });</pre>
Выбор	
<pre>\$i = 5; switch (\$i) { case 0: echo "i равно 0"; break; case 1: echo "i равно 1"; break; default: echo "i больше 1"; }</pre>	<pre>var i = 5; switch (i) { case 0: alert("i равно 0"); break; case 1: alert("i равно 1"); break; default: alert("i больше 1"); }</pre>

Массивы	
<pre> \$arr = array(1, 2, 3, 4); \$arr = array("name" => "user", "pass" => "0000",); // Начиная с PHP 5.4 \$arr = ["name" => "user", "pass" => "0000",]; \$matrix = array(0 => array(1,2), 1 => array(1,2), 2 => array(1,2),); </pre>	<pre> var arr = new Array(1, 2, 3, 4); var arr = [1, 2, 3, 4]; var arr = []; arr["name"] = "user"; arr["pass"] = "0000"; var matrix = [[1,2],[3,4],[5,6]]; var arr = [1, {x:1, y:2}}, 2, {x:3, y:4}}]; </pre>
Объекты	
<pre> class MyClass { function myFunc() { echo "Test"; } } \$obj = new MyClass; \$obj->myFunc(); </pre>	<pre> var point = {x:0, y:0}; var book = { "main title": "JavaScript", 'sub-title': "Pocket Ref" }; var obj = new Object(); var obj = Object.create({x:1, y:2}); var obj = { x: 1.0, get funcValue() { return this.x; }, set funcValue(value){ this.x = value; } }; </pre>
Функции	
<pre> \$a = 25; function myfunc(\$arg1, &\$arg2) { \$arg2 = 0; return \$arg1 * 2; } myfunc(5, \$a); </pre>	<pre> var a = 25; function myfunc(arg1, arg2) { window.a = 0; return \$arg1 * 2; } myfunc(5, a); var point = function(arg1, arg2) { window.a = 0; return \$arg1 * 2; } </pre>
Анонимные функции	
<pre> echo function () { return 'Hello World!'; } </pre>	<pre> alert(function () { return 'Hello World!'; }); </pre>

Классы

<pre>class MyClass { public \$value = 'значение'; public function myFunc() { echo \$this->value; } } \$c = new MyClass(); \$c->myFunc();</pre>	<pre>function MyClass(value) { this.value = value; } MyClass.prototype.myFunc = function() { return this.value; }; var c = new MyClass('текст'); c.myFunc();</pre>
<pre>class MyClass { public \$value = 'значение'; public static function myFunc() { echo \$this->value; } } MyClass::myFunc(); MyClass::value = 'текст'</pre>	<pre>function MyClass() { this.value = 'значение'; this.myFunc = function() { return this.value; }; } var c = new MyClass(); c.myFunc();</pre>
<pre>class BaseClass { function __construct() { print "Конструктор класса"; } } class SubClass extends BaseClass { function __construct() { parent::__construct(); print "Конструктор класса"; } }</pre>	<pre>function MyClass(value) { var class = Object.create(MyClass.methods); class.value = value; return class; } MyClass.methods = { myFunc: function() { return this.value; } }; var c = MyClass('текст'); c.myFunc();</pre>

Наследование

<pre>class MyClass1 { //... } class MyClass2 extends MyClass1 { //... }</pre>	<pre>function MyClass1() { //... } MyClass1.prototype.func = function(){} function MyClass2() { //... } MyClass2.prototype = Object.create(MyClass1.prototype);</pre>
--	--

Модули / Пространства имен

<pre>namespace MyProject; const CONNECT_OK = 1; class Connection { } function connect() { }</pre>	<pre>(function () { var CONNECT_OK = 1; function connect() { //... } })();</pre>
---	--

	<pre> var MyProject = (function() { return { CONNECT_OK: 1, connect: connect() { } } })(); </pre>
--	--

События

	<pre> document.getElementById("myButton"). addEventListener("click", onClick); document.getElementById("myButton"). removeEventListener("click", onClick); function onClick(){} window.onload = function() { alert('Документ загружен'); }; </pre>
--	--

Ошибки

<pre> // Выключение протоколирования ошибок error_reporting(0); try { //... } catch (Exception \$e) { echo \$e->getMessage(); } finally { //... } </pre>	<pre> try { //... } catch (er) { document.write(er); } try { //... } finally { //... } </pre>
--	--